

1

REMASTERING BY EMULATION

FIELD

The present application relates to technically inventive, non-routine solutions that are necessarily rooted in computer technology and that produce concrete technical improvements.

BACKGROUND

As understood herein, remastering some entertainment assets such as older computer games to be played on later model consoles can entail creating an entirely new software to accommodate improved capabilities of the later model console. This is time consuming an expensive.

SUMMARY

As further understood herein, the original (legacy) software code of an entertainment asset such as a computer game designed to be played on a relatively lower powered display can be preserved for use with model consoles with improved capability displays, while effecting remastering of the presentation through emulation principles.

Thus, computer game remastering is intended herein, but it is to be understood that present principles are not limited to computer games.

Accordingly, a device includes one or more computer memories that are not transitory signals and that include instructions executable by at least one processor to intercept calls for textures from computer game software. The instructions are executable to, for at least a first intercepted call for a first texture, execute a hash on information associated with the first texture to render a result. The instructions are executable to, using the result, retrieve from a data structure a remastered texture responsive to the remastered texture being associated with a remastered texture hash from a data structure matching the result, and to insert the remastered texture into presentation of a computer game on the fly as the game is presented under control of the computer game software.

In some examples the computer game software is a legacy software designed for use on a first display having first resolution, and remastered textures are established for presentation on a second display having a second resolution higher than the first resolution without changing code in the legacy software.

In example embodiments the instructions may be executable to execute the hash on plural of the textures subject to respective intercepted calls. If desired, the instructions may be executable to, responsive to the result not matching any remastered texture hash in the database, present an original texture responsive to the first intercepted call.

In non-limiting embodiments the instructions may be executable to intercept calls for legacy audio from computer game software, retrieve, from a data structure, a remastered version of the legacy audio, and insert the remastered version into presentation of the computer game on the fly. The first intercepted call may be for an audio track or an audio sample.

Additionally, optional instructions can be executable to intercept calls for three dimensional (3D) geometry from computer game software, and for at least a first intercepted call for 3D geometry, execute a hash on information in an associated 3D geometry to render a 3D geometry result. These instructions may be executable to, using the 3D

2

geometry result, retrieve from a data structure a remastered 3D geometry responsive to the remastered 3D geometry being associated with a remastered 3D geometry hash from a data structure matching the 3D geometry result, and insert the remastered 3D geometry into presentation of the computer game on the fly.

The device may include the processor and/or a display for presenting the computer game.

In another aspect, an apparatus includes at least one computer storage comprising instructions executable by at least one processor, and at least one processor configured to access the instructions for intercepting, from a presentation control, signals for retrieving assets. For at least some intercepted signals, the processor executing the instructions associates respective unique identifiers with respective assets associated with respective intercepted signals to render respective results. For each result, the processor accesses a data structure to locate a match, and responsive to locating a match of a result, retrieves an associated modified asset for presenting the associated modified asset on a display according to the presentation control.

A method according to the operation of the apparatus also is provided.

In another aspect, a device includes one or more computer memories that are not a transitory signals and that include instructions executable by at least one processor for intercepting calls for assets from computer game software, and for at least a first intercepted call for a first asset, associating a first unique identifier with the first asset. The instructions are executable for determining whether the first unique identifier matches an entry in a remastering data structure, and responsive to the first unique identifier matching an entry in the data structure, continuing to intercept the calls. However, responsive to the first unique identifier not matching an entry in the data structure, the first unique identifier and the first asset are stored in the data structure. The instructions are executable for outputting the first asset to a remastering computer. A remastered asset is received from the remastering computer, with the remastered asset being a higher resolution version of the first asset, and the remastered asset is associated, in the data structure, with a unique identifier.

A method according to the operation of the device also is contemplated.

The details of the present application, both as to its structure and operation, can best be understood in reference to the accompanying drawings, in which like reference numerals refer to like parts, and in which:

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of an example system including an example in accordance with present principles;

FIG. 2 is block diagram of an example system;

FIG. 3 is a flow chart of example logic for creating remastered game textures;

FIG. 4 is a flow chart of additional example logic for creating remastered game textures;

FIGS. 5-7 are screen shots showing an example remastering operation;

FIG. 8 is a flow chart of example logic for presenting remastered textures into a legacy computer game on the fly; and

FIG. 9 is a flow chart of example logic for presenting remastered audio.